ACTS Tracking For Muon Collider

Karol Krizka

on behalf of the Muon Collider Detector and Physics Group

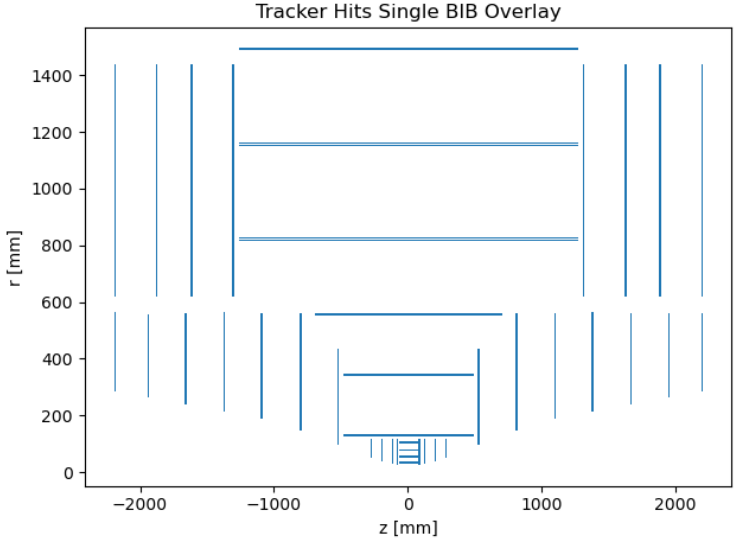
April 20, 2021

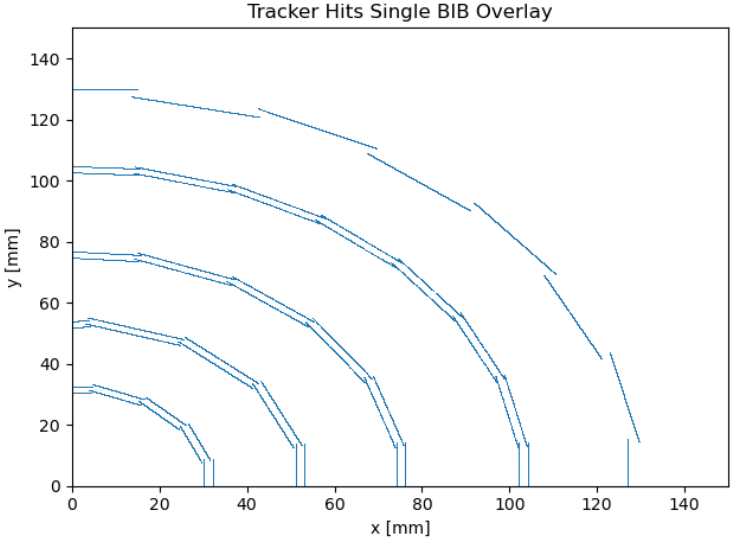
APS April Meeting

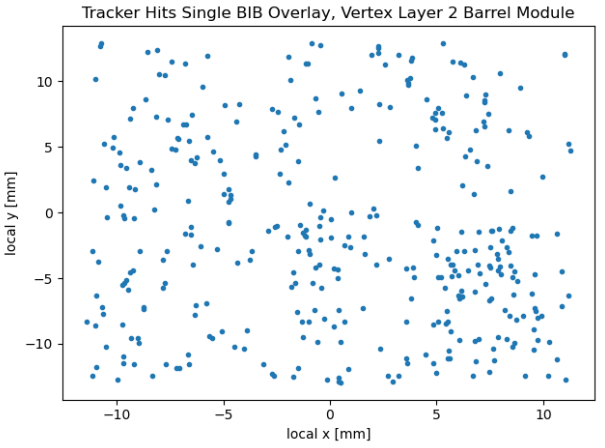
Current Tracking Implementation

More Information:
- Detector overview from Simone
- Tracking overview from Massimo

- **Designed for the e^+e^- environment**
  - Inherited as part of the CLIC software framework
- **Implements conformal tracking (1908.00256)**
  - Transform circular tracks into straight lines using conformal map
  - Use *cellular automata* to look for lines, allowing for deviations
- **Problem: $\mu^+\mu^-$ collider is much busier due to Beam Induced Bkg**
  - Heavy pre-filtering of hits is necessary for conformal tracking to work







A Common Tracking Software

ACTS GitHub

ACTS is a generic library for track fitting at collider experiments.

- **Dedicated team working on advancing tracking algorithms**
  - Tracking is hard!
- **Allows us explore alternate algorithms**
  - Triplet-based seeding optimized for high multiplicity environments
- **Code optimization come for free**
  - Also explores modern computing architectures (ie: CUDA)

HOWTO: Tracking

1)Pattern recognition

- Create collection of hits corresponding to track candidates
- ilcSoft: Conformal tracking
- ACTS: Triplet-based seeding + Combination Kalman Filter

2)Track fit

- Kalman Filter to obtain track parameters
- Material description of detector required
- Separate implementations ilcSoft and ACTS

Following studies reconstruct a single muon with $p_T \in [0.5,10]$ GeV.

Truth Tracking

Validates detector description in ACTS.

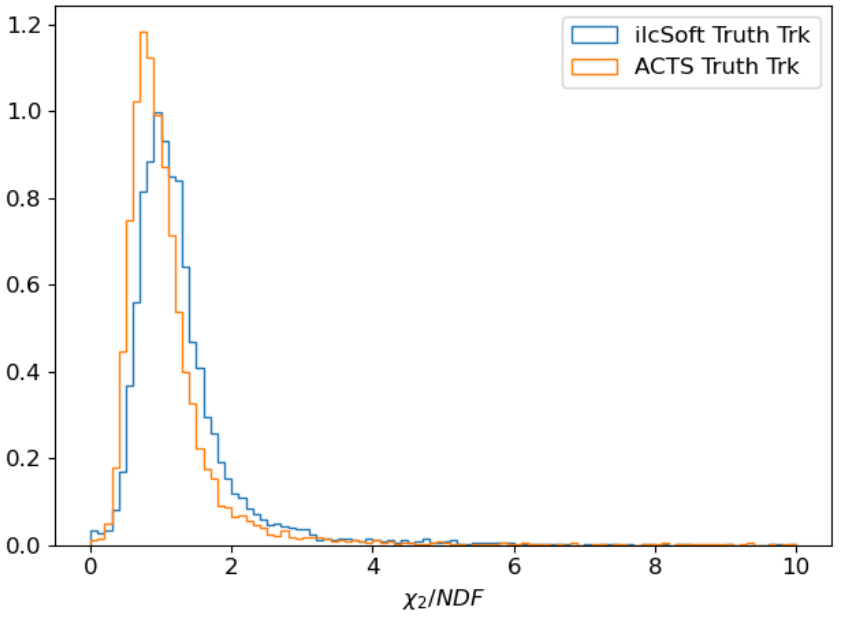
Pattern Recognition

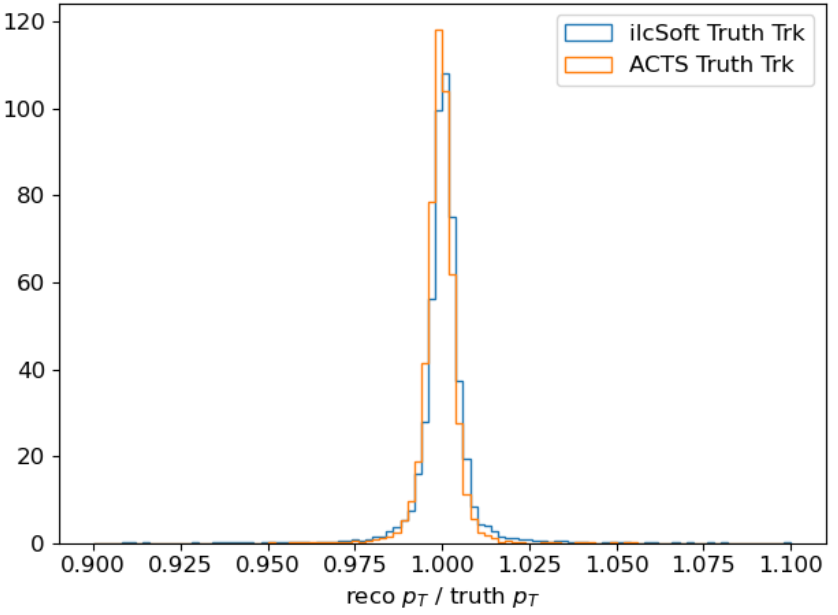
- Use hits associated to MC particle (100% efficiency)
- Same code for ilcSoft and ACTS

Track Fit

- Kalman Filter, but ACTS vs ilcSoft implementation

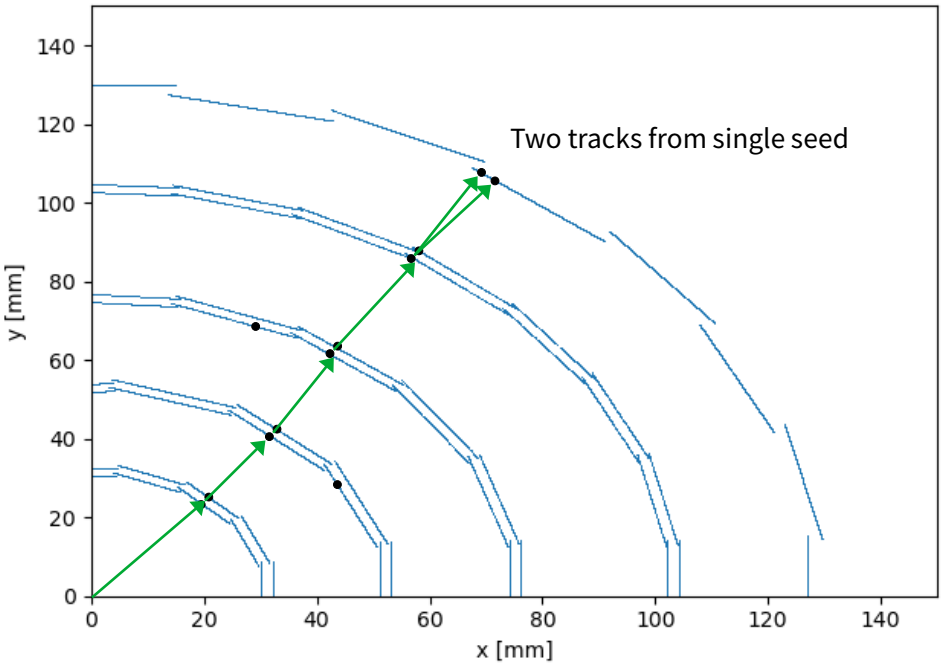
Fit Library	Execution Time
ACTS	0.5 ms / evt
ilcSoft	100 ms / evt





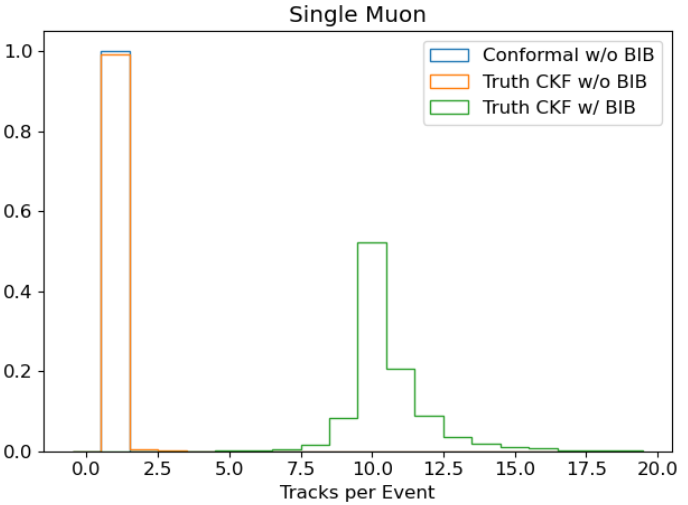
Combinatorial Kalman Filter

1) **Start with an estimate of track parameters**
   - ie: from seeding stage
2) **Propagate track to next layer**
3) **Look for compatible hits**
4) **Update track with new hit**
   - Multiple compatible hits → create multiple tracks
5) **Repeat steps 2)-5) with all track parameters until last layer**
6) **Refit all resulting tracks**



Note: Doublet layers are currently treated as individual layers.

Truth CKF Tracking



Seeding (the truth part)

- Use MC particle kinematics

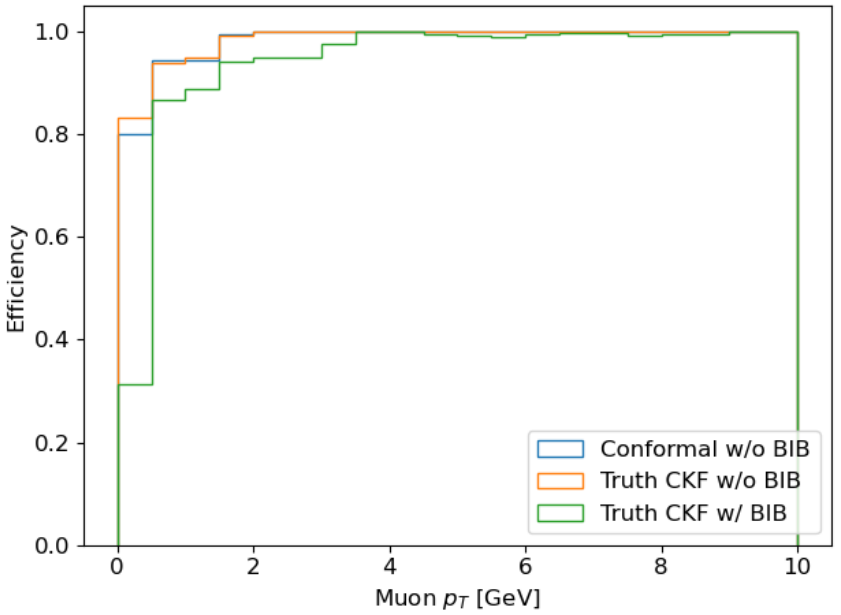
Track Fit

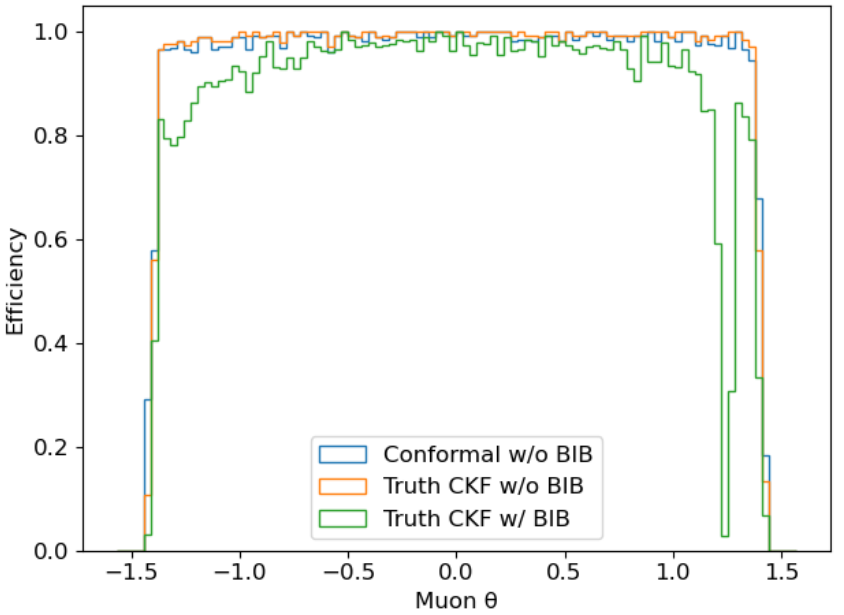
- Combinatorial Kalman Filter in ACTS

Fit Library	Execution Time
Conformal	120 ms / evt
ACTS	0.5 ms / evt
BIB + ACTS	5 s / evt

Overlap Removal

- Group by tracks sharing 50% of the hits, pick one with most (or highest χ^2)





Triplet Seeding

1)Choose *N* layers for seeding

- N=4 in our case

2)Form seeds containing three hits

- All possible combinations in *N* layers

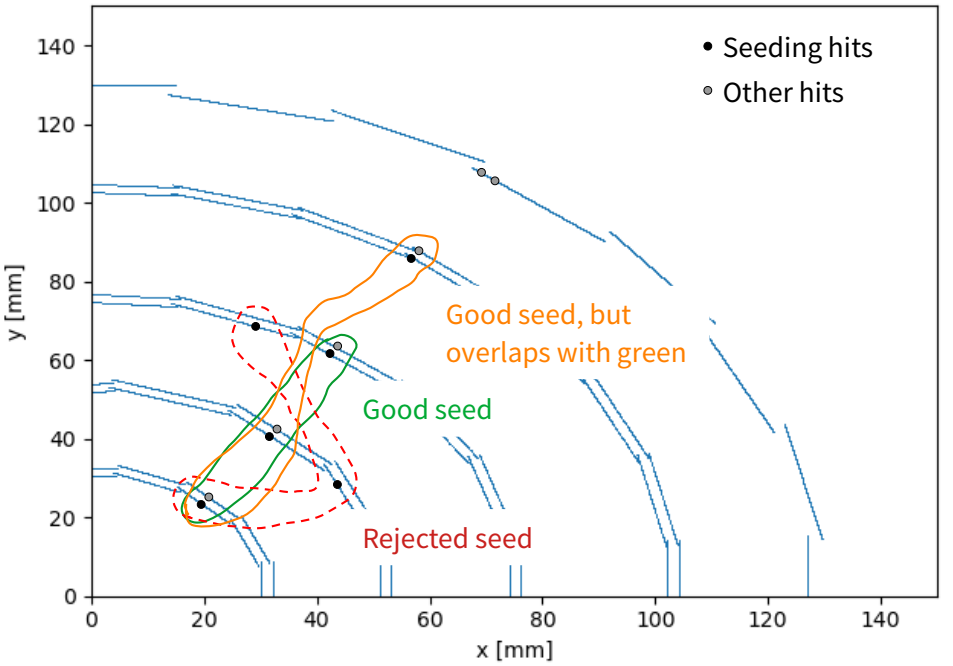
3)Remove bad seeds

- Based on compatibility with helix

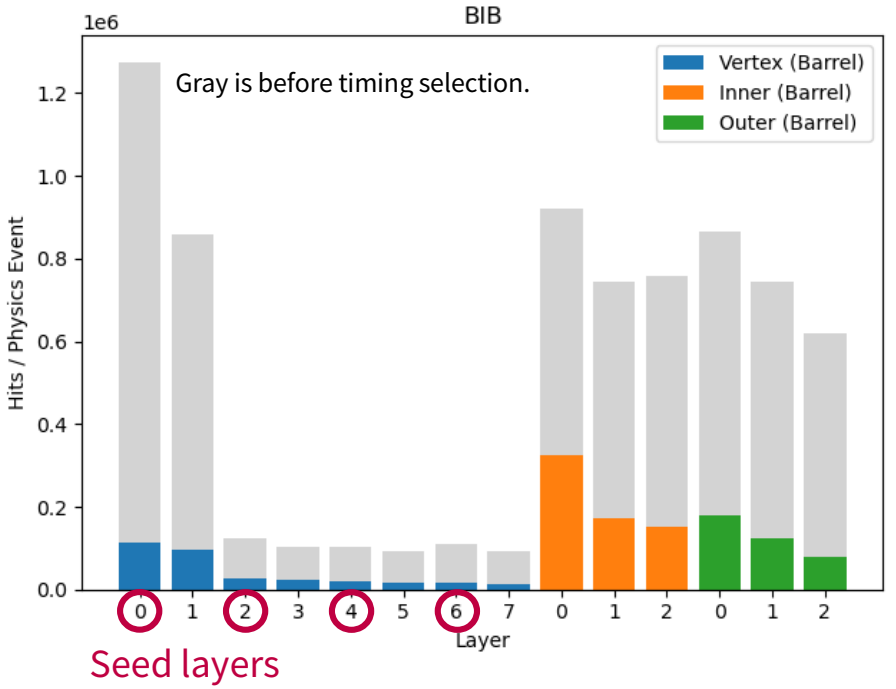
4)Remove overlap between seeds

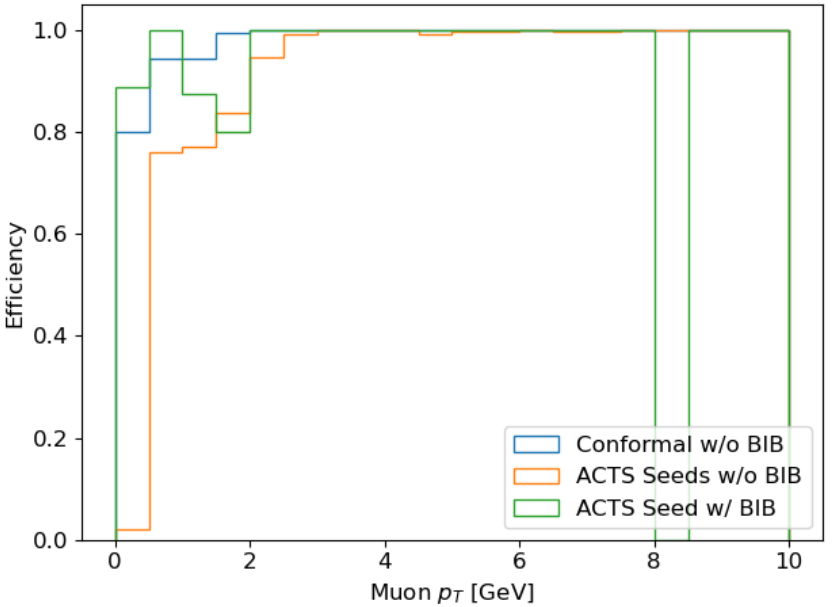
- Based on middle hit in seed

5)Use estimated track parameters as input to CKF



Track Seeding





- **Using only inner part of the Vertex doubles**
  - Prevents redundant "too close together" combinations
- **~350k seeds per event**
  - 200 ms* / seed x 350k seeds / event = ~20 hours / evt
  - Compare with 1 week / evt in conformal tracking

	Combinations
All Triplets	700B
Seeds	2000

Per region, x144 regions

* CKF on "wrong" seed is faster than on a "true" seed.

Towards Seeded CKF

- **Need to reduce number of seeds by at least x10**
  - Reduce hits via cluster shape analysis
  - Tighter seed “helix compatibility” requirements
  - Consistent timing of hits within a triplet
  - Consistent hits within doublet layers
- **Need to recover seed efficiency at low p_T**
  - Optimization of seed finding configuration

Conclusions

- **Current baseline for tracking is conformal tracking**
  - Found to be sub-optimal in the μ+μ- environment
- **Tried to use algorithms from the LHC experiments**
  - Triplet-seeding + combinatorial kalman filter
  - Implemented using the ACTS library
- **Out-of-the-box: BIB is too much even for triplet seeding**
  - O(100k) seeds → 1 day / per event
  - Still a lot existing, but unused, handles
- **ACTS implementation of common algorithms is faster**

BACKUP

Context

my session: Y07

More Details

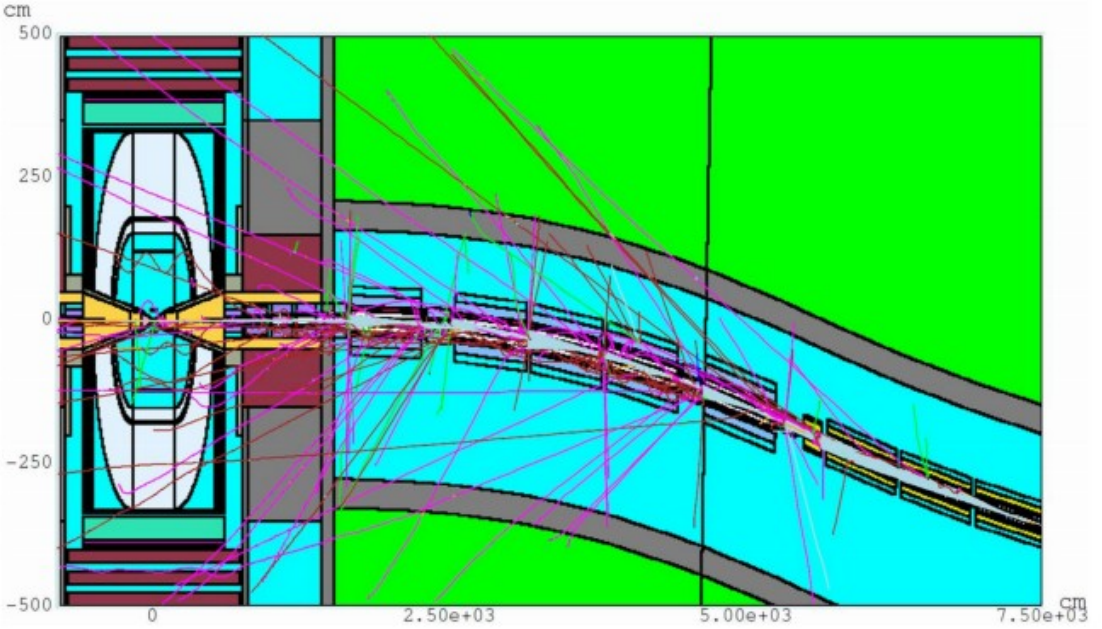
- **Part of Muon Collider Symposium at APS, 10+2 talk**
- **At the very end, but mostly theory talks in my session**

Beam Induced Background (BIB)

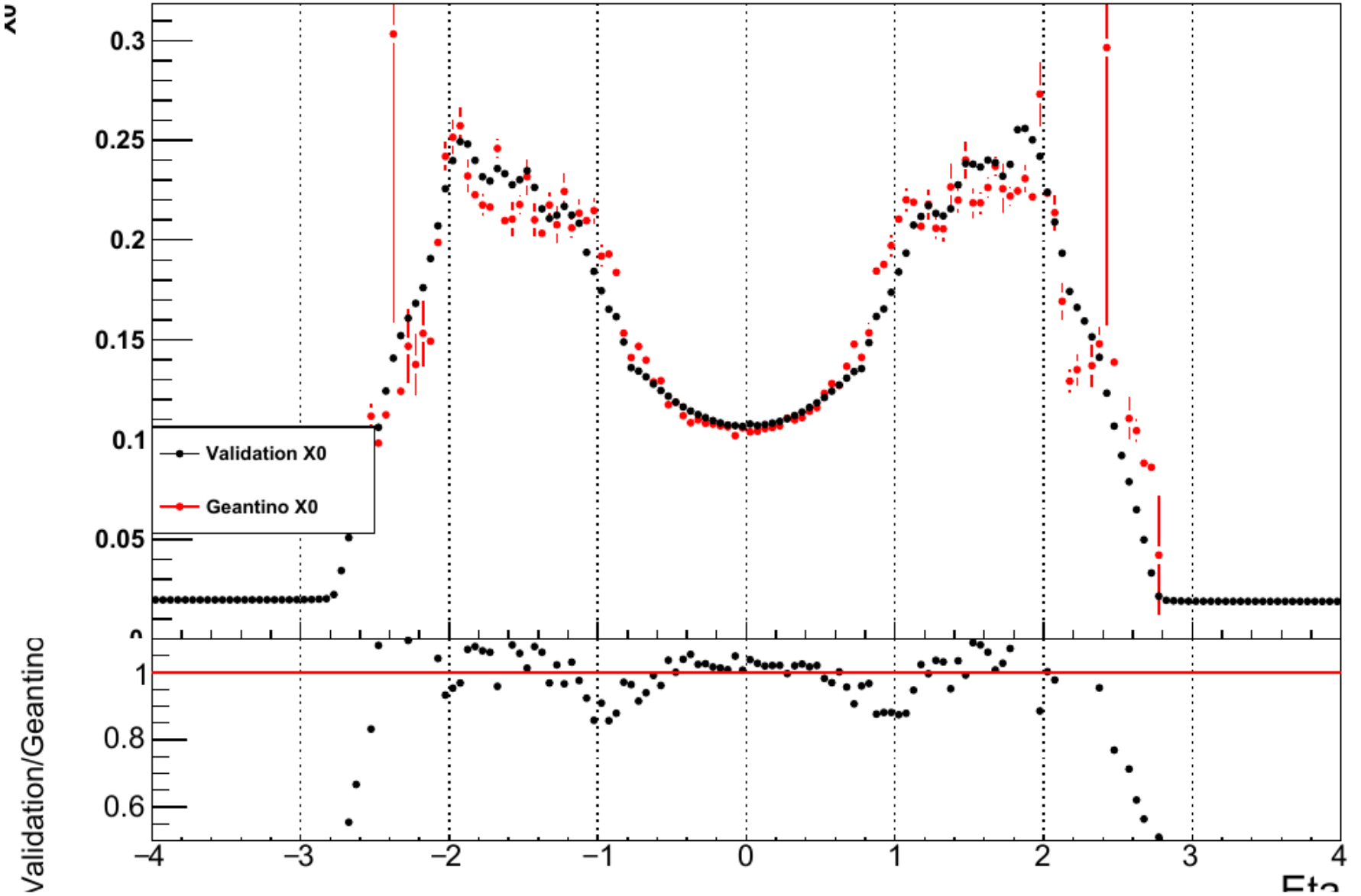
- Muon decay products from the beam striking the detector
- Somewhat shielded with “nozzle”, but multiplicity still large
- Precise timing in detector will be important

Tracker

- Vertex is made up of doublet Si layers
  - 20x20 μm pixels, 50ps time resolution
- Remainder of tracker is single layer Si
  - 50x50 μm pixels, 100ps time resolution

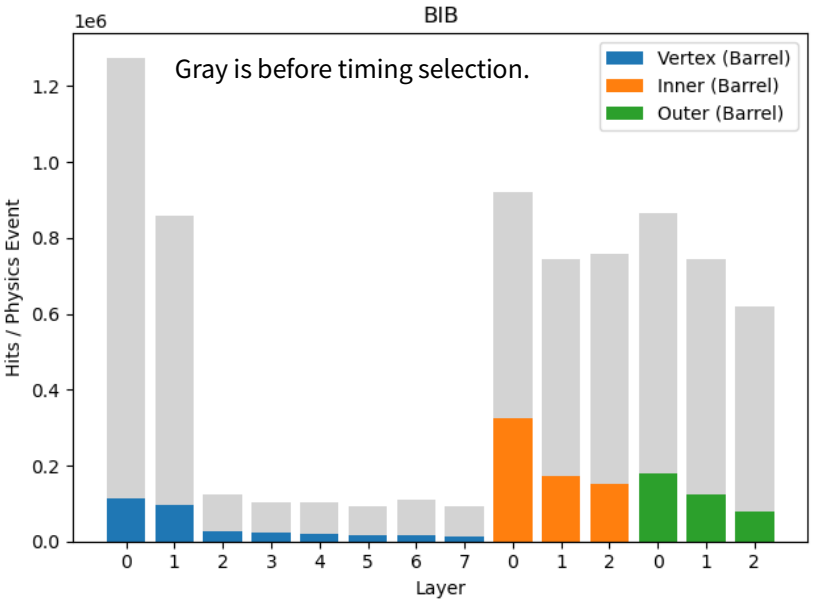


Material Validation

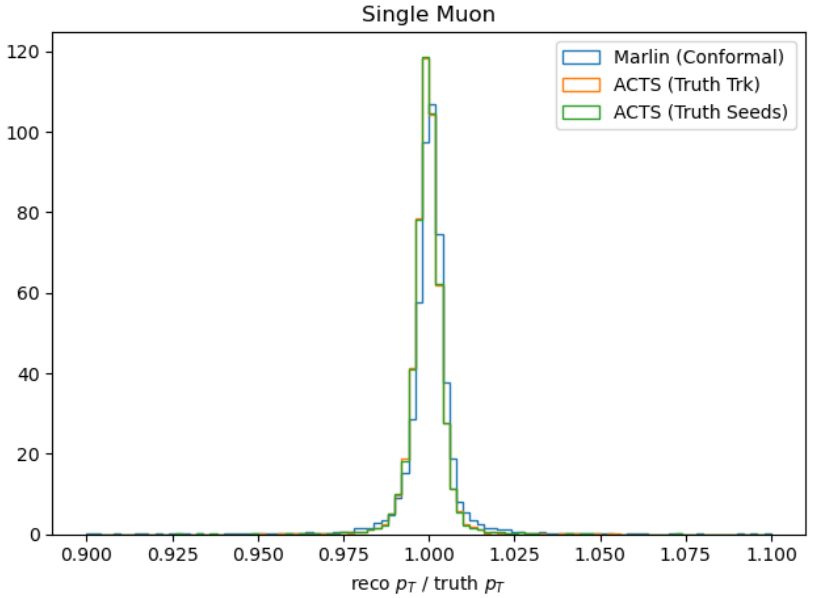


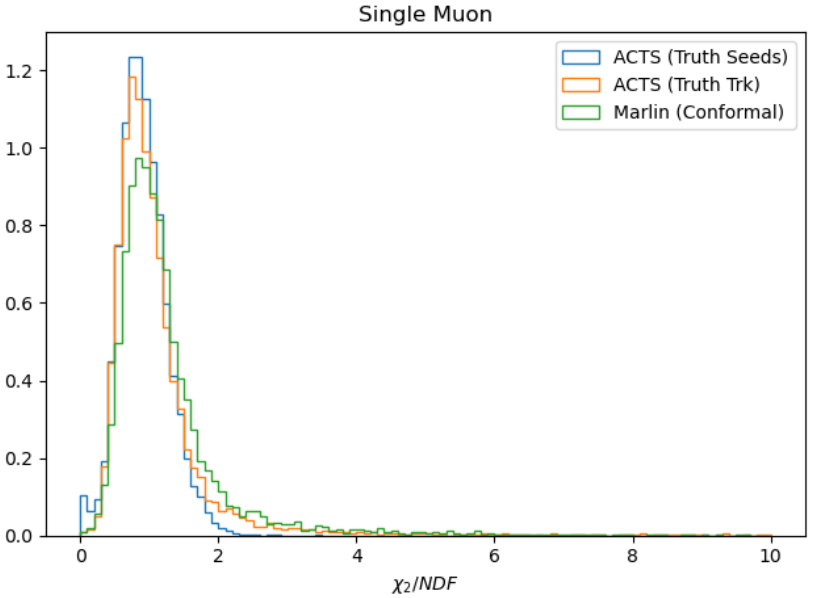
Add notes about importing MCC geometry into ACTS.

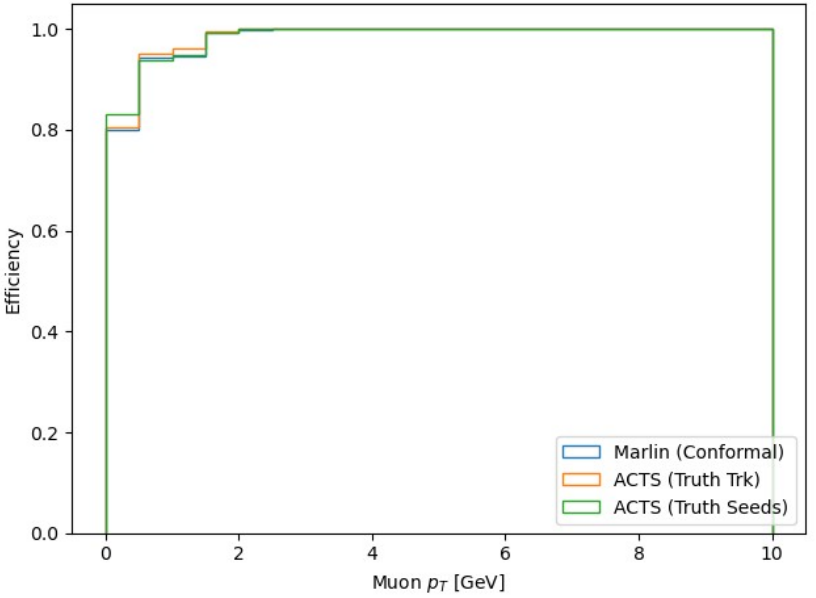
Hit Multiplicity

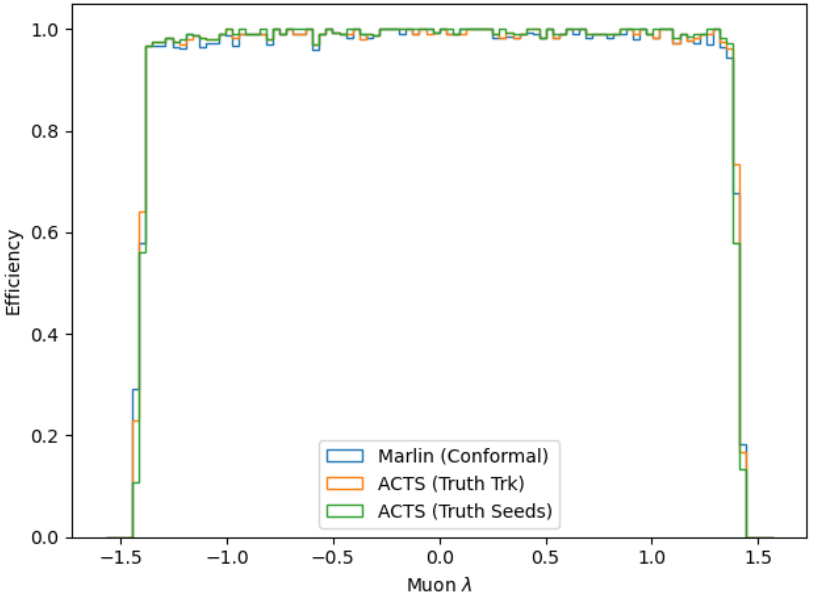


More Truth CKF



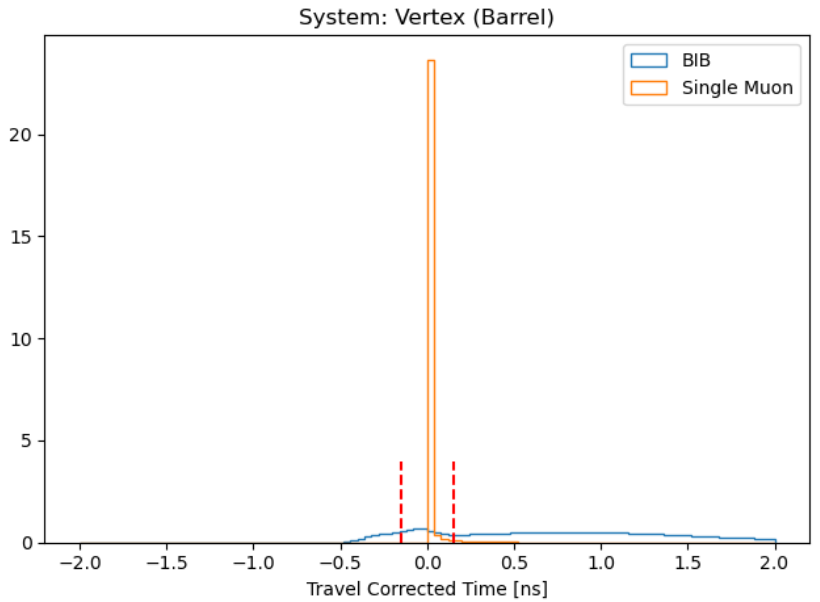


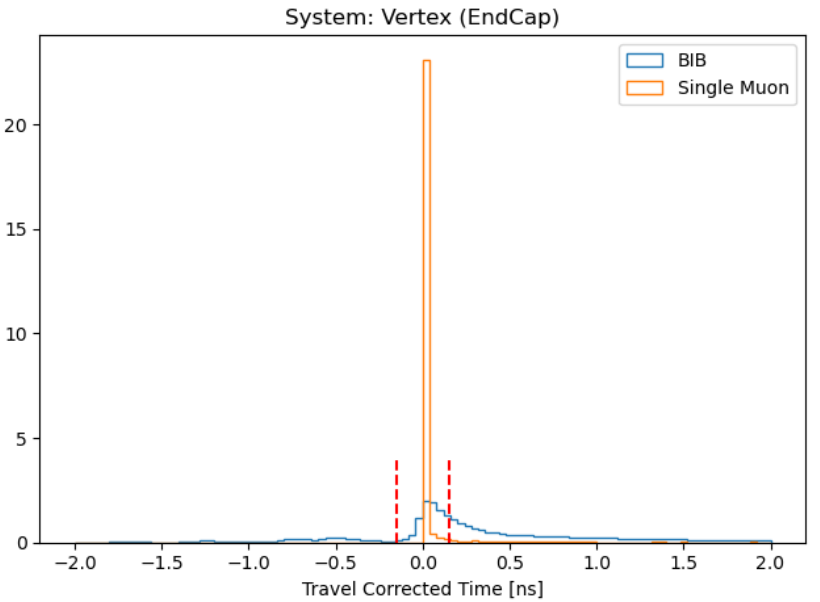




BIB Timing

SimTrackerHit (no smearing, usually 50 ps)

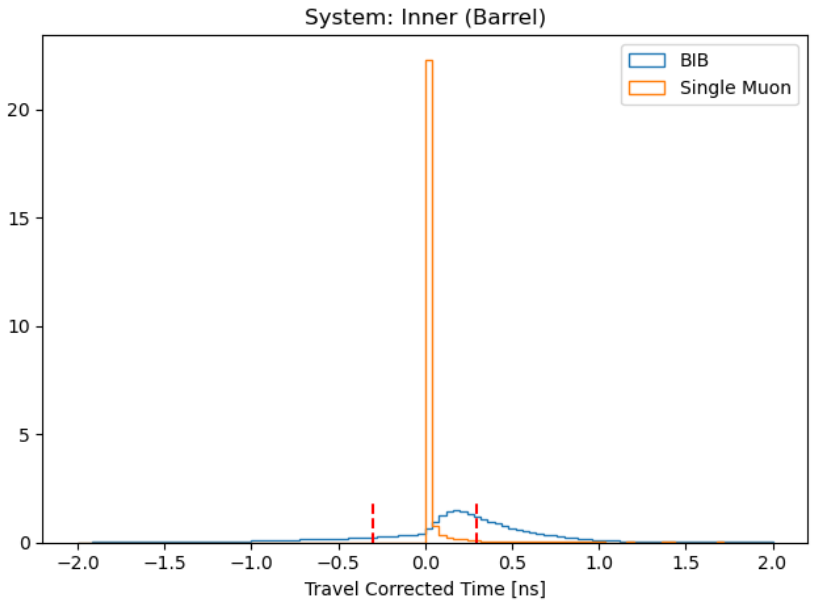


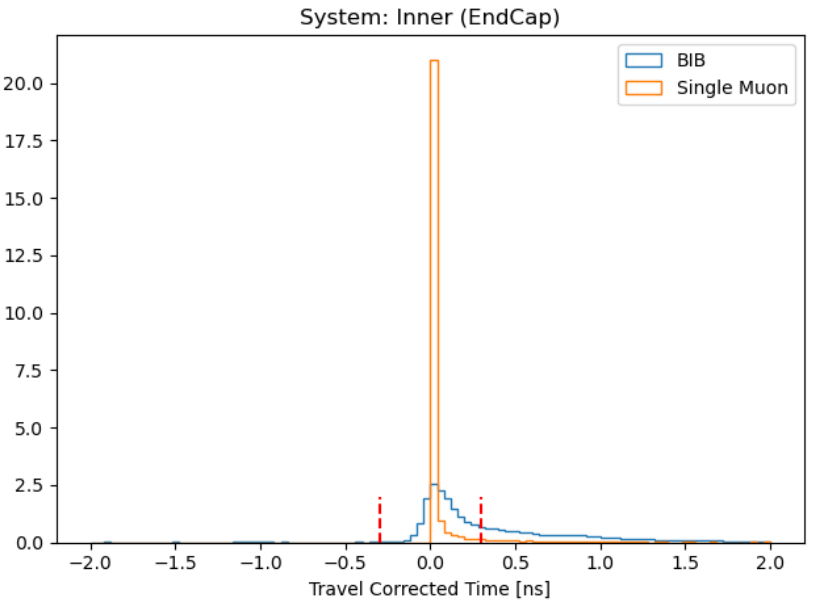


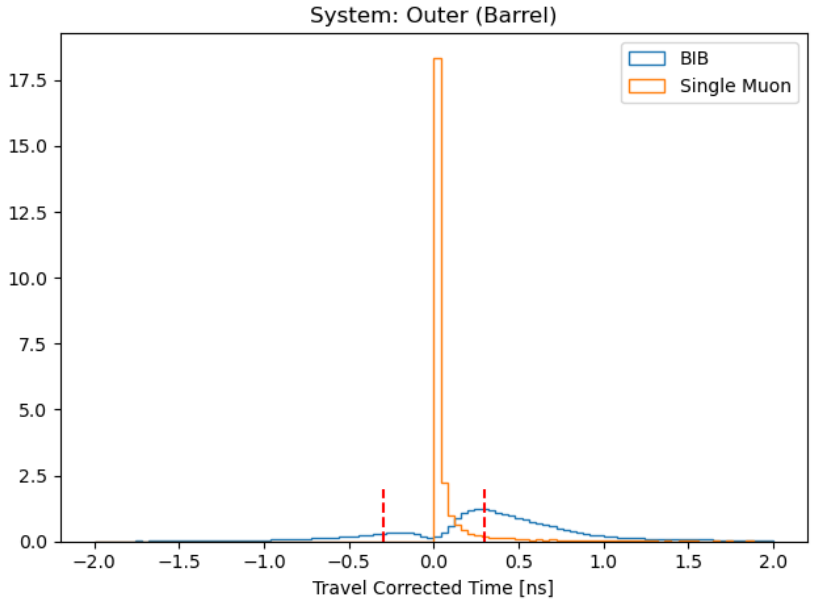
- **Based on SimTrackerHit (no smearing)**
  - Current default is 50 ps time resolution
- **Does not include cuts from Overlay processor**

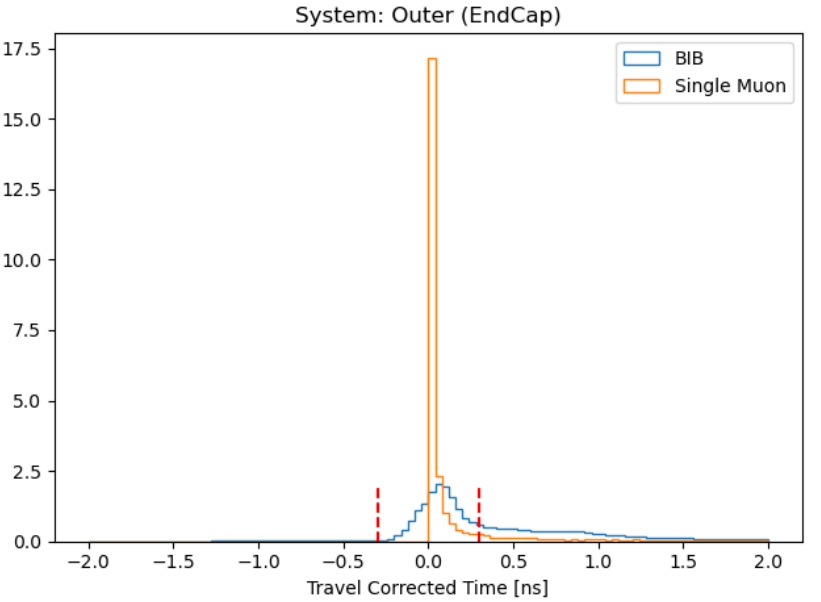
BIB Timing

SimTrackerHit (no smearing, usually 50 ps)

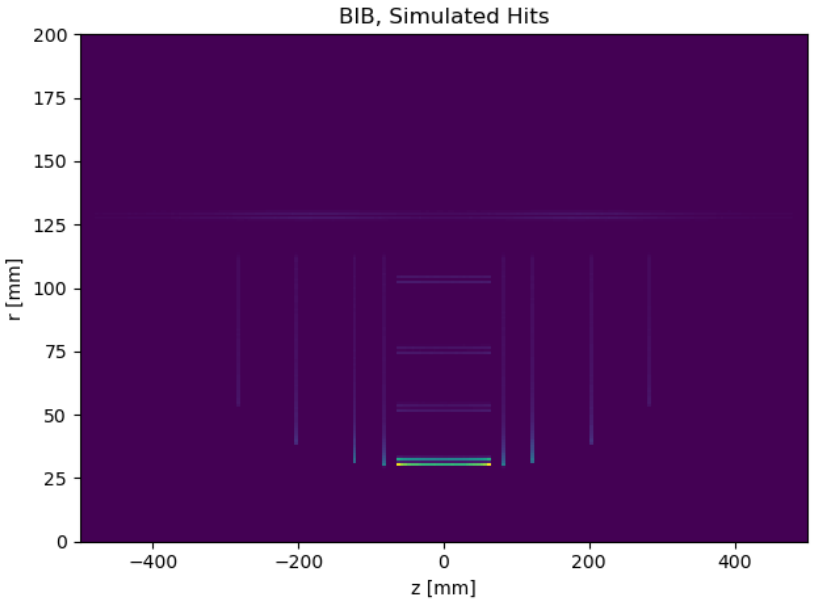


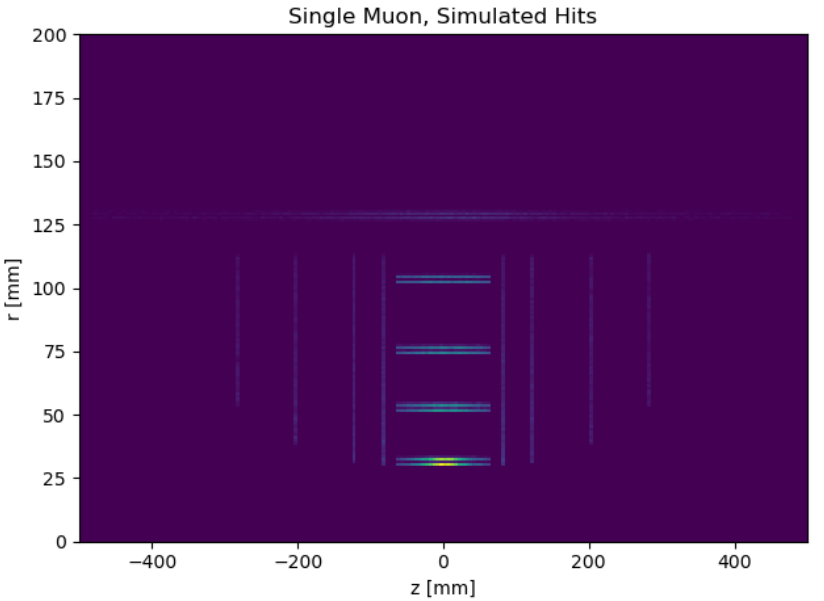


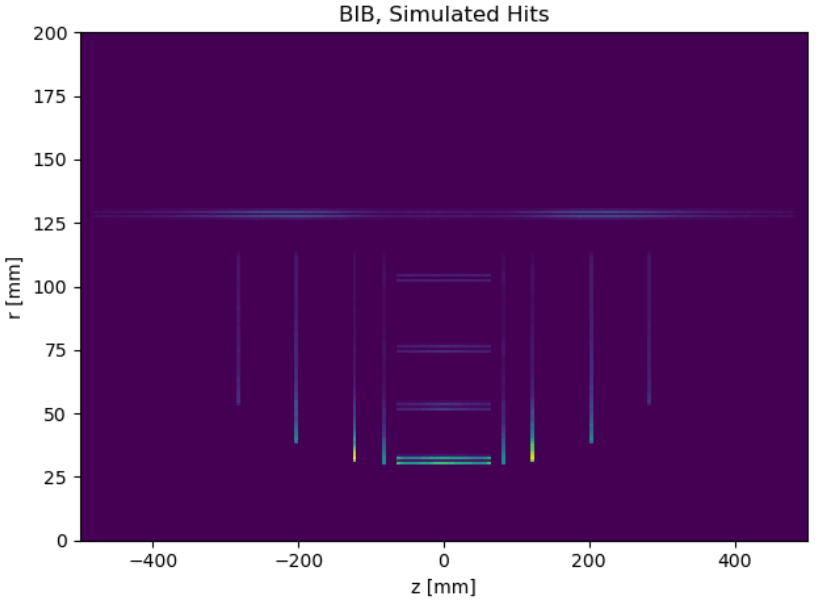


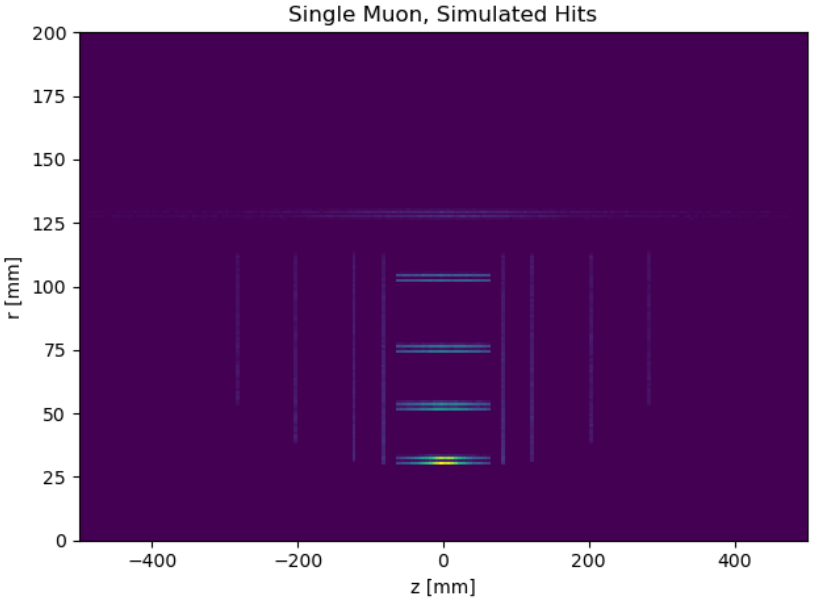


BIB Distribution



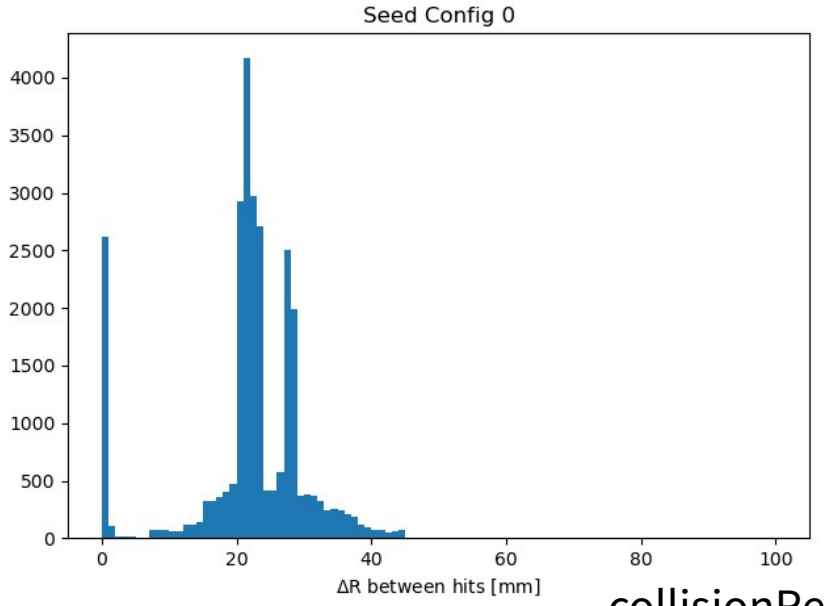




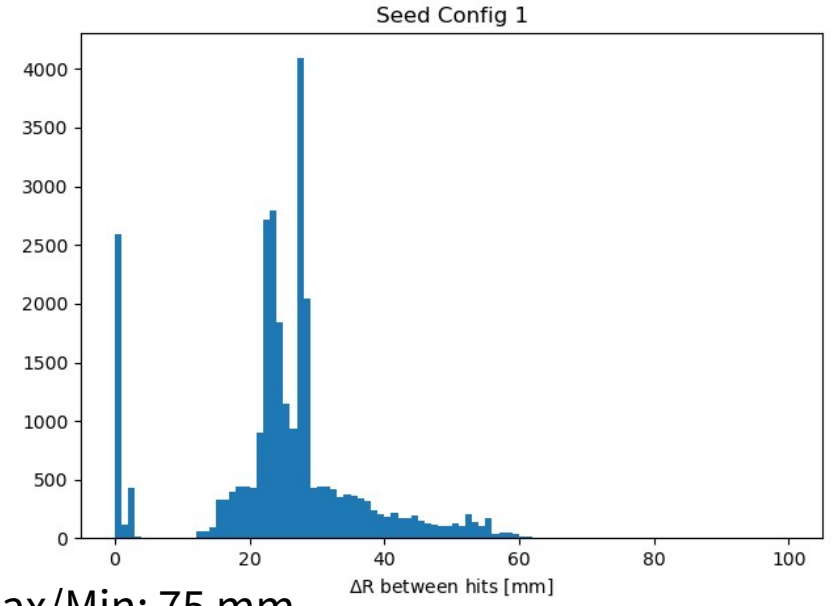


Optimizing Seeding Settings

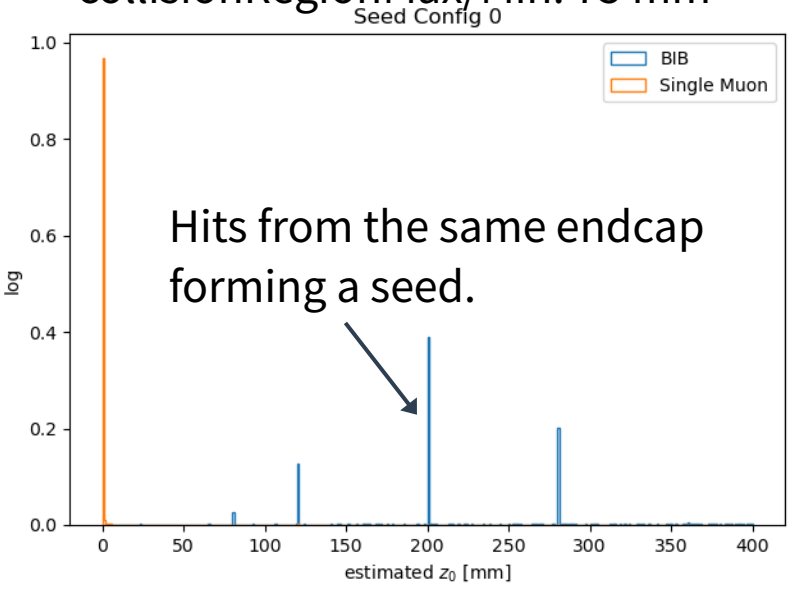
deltaRMin: 5 mm to remove same layer



deltaRMax: 80 mm



collisionRegionMax/Min: 75 mm



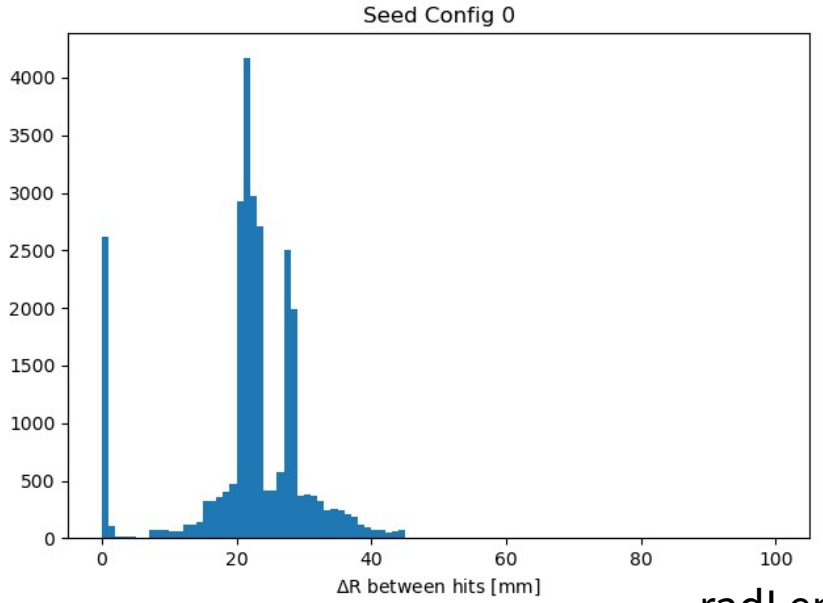
Hits from the same endcap forming a seed.

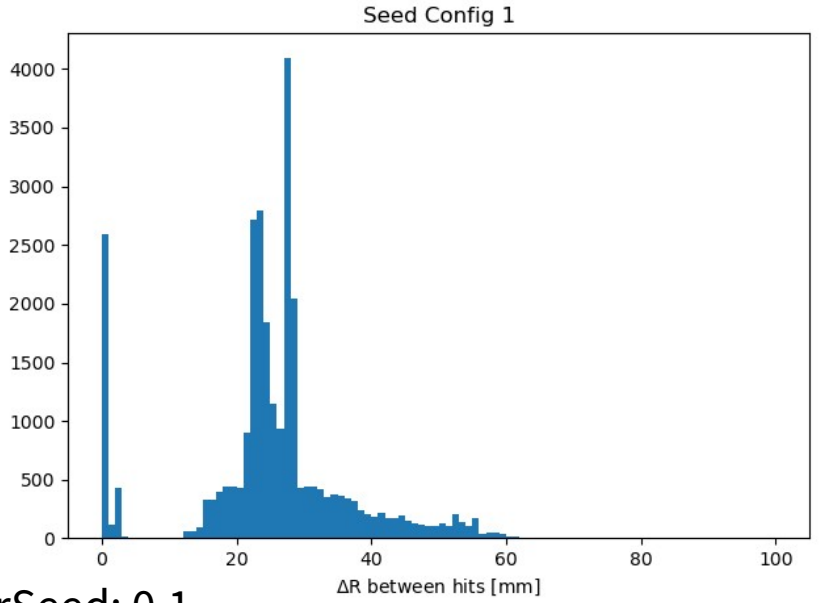
Bunch length:
5 mm to 10 mm
Maybe try 30 mm?

Need to keep collisionRegion cuts loose to allow for displaced tracks

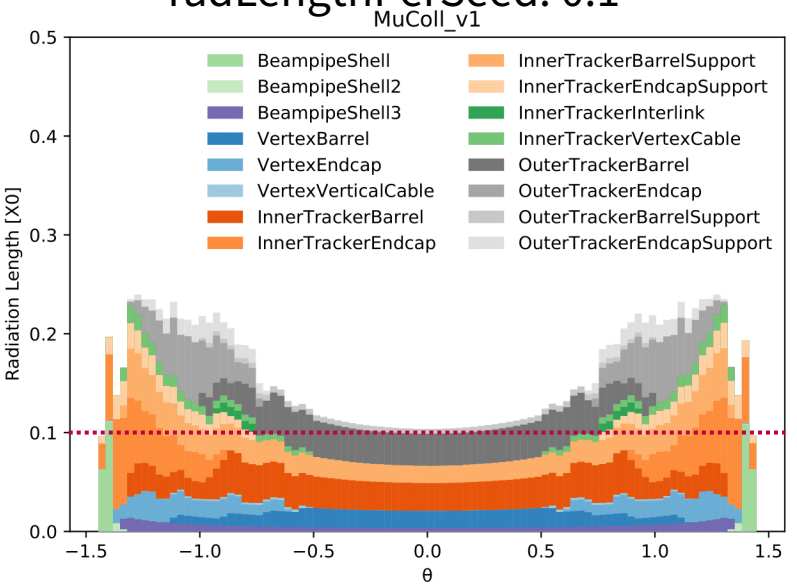
Optimizing Seeding Settings

deltaRMax: use 80 for both

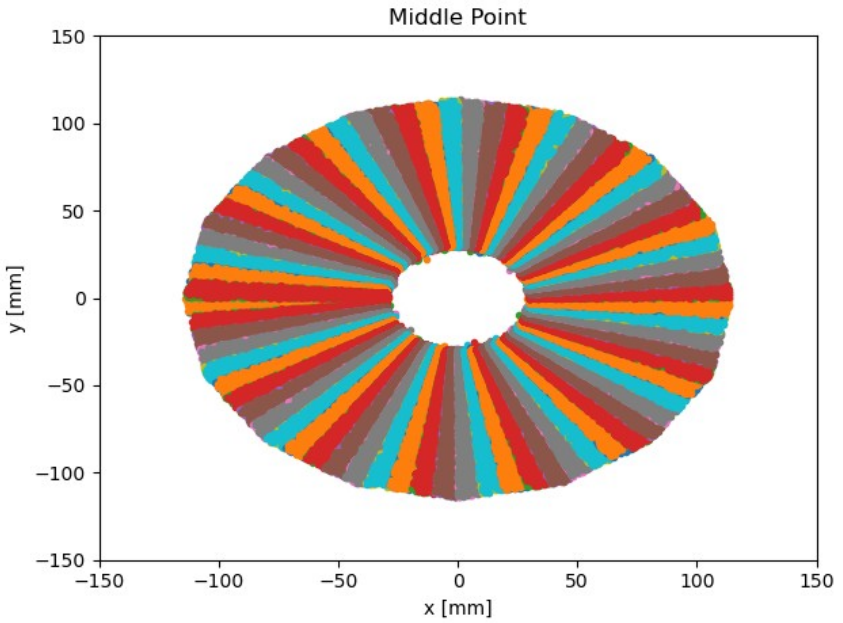


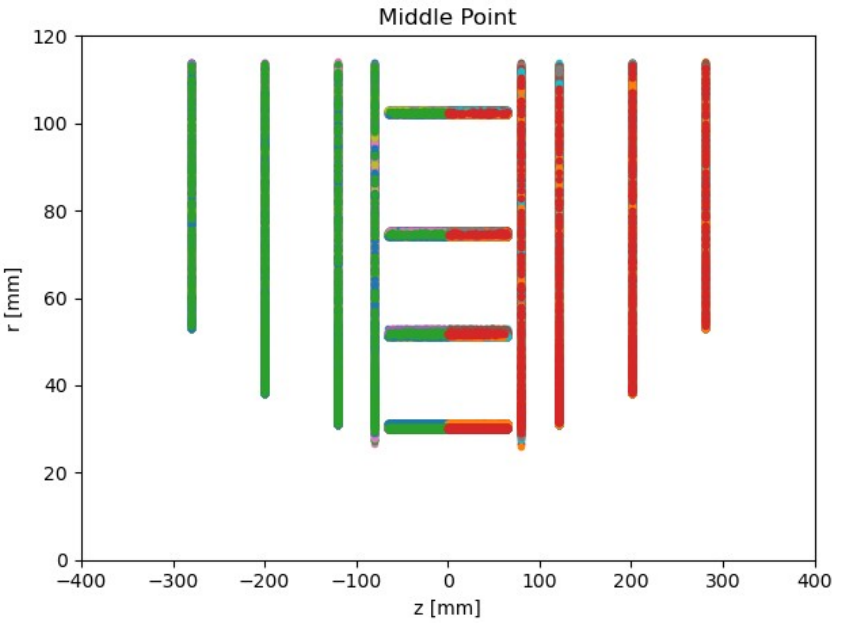


radLengthPerSeed: 0.1

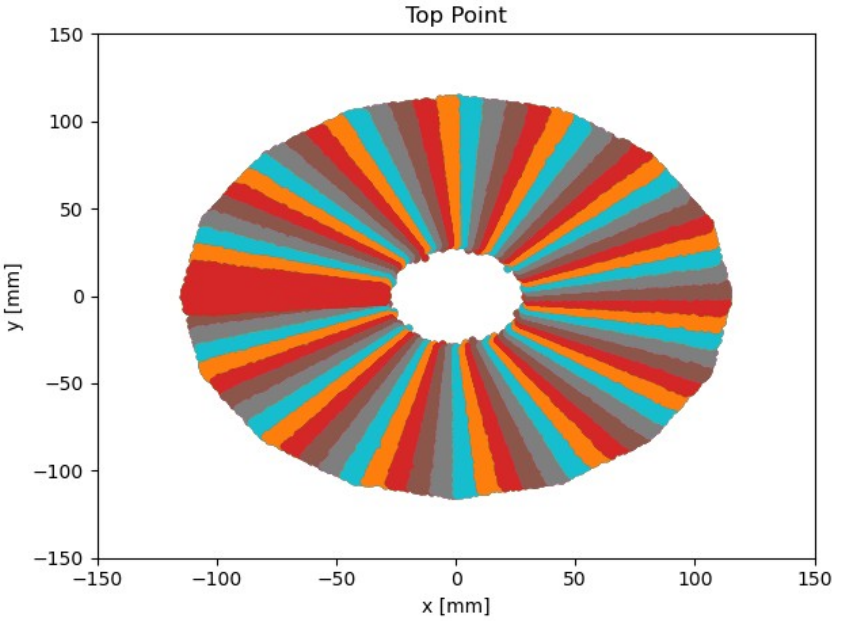


Seed Groups (Cfg 0)





- **ACTS looks for seeds in overlapping groups (binning)**
  - Middle point is binned in z (2) and ϕ (72)
  - Top/Bottom points are binned more coarsely (and overlap) in ϕ only
  - Top/Bottom bins seem to be identical
- **How is the size of top/bottom bins set?**



Combinations in Each Group (with BIB)

	Config 0	Config 1
Top	16278	25536
Middle	2745	4227
Bottom	16278	25536
Comb	700B	3T
Rd Comb	800M	1.1B
Seeds	2000	2000

1) O(trillion) combinations in each group

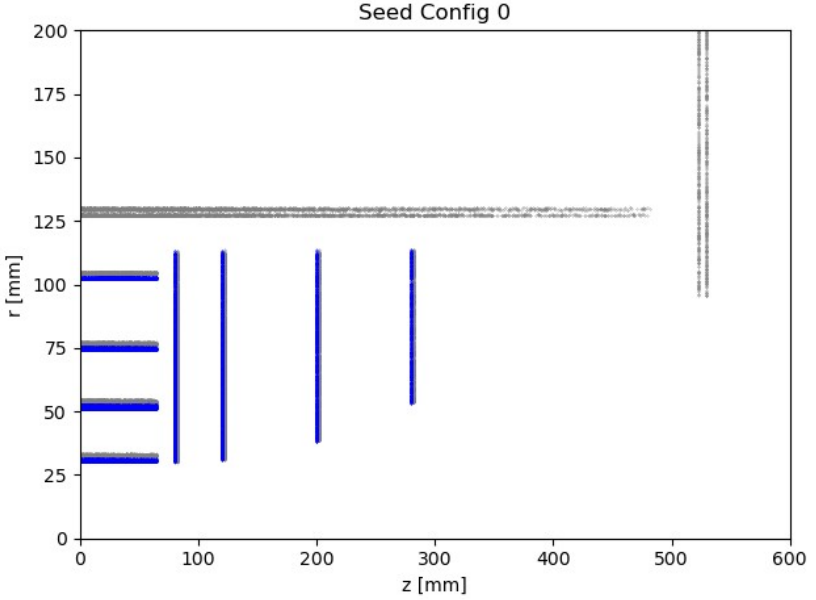
2) O(1 billion) possible seeds after initial geometry cuts

3) O(1000) final seeds after helix estimate and overlap removal

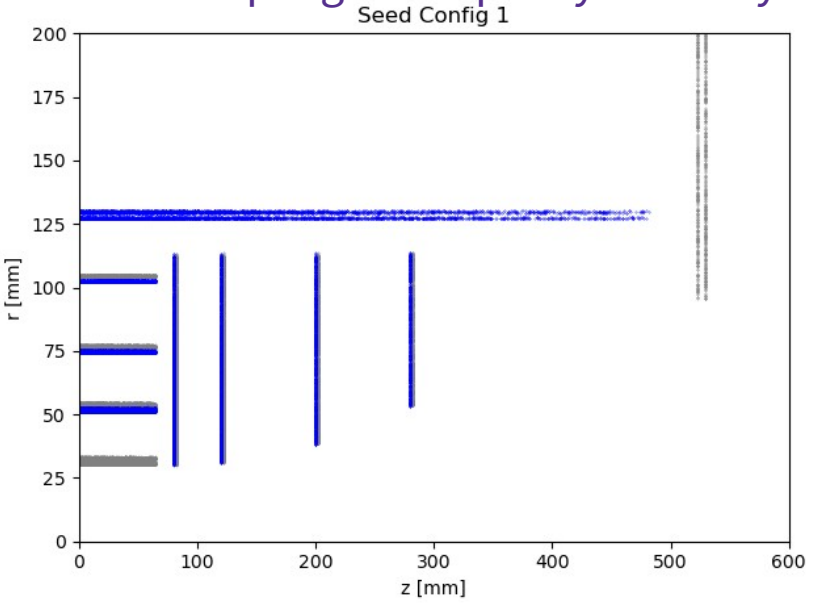
- This is the slowest step

Seeding Layers

Seed 0: Vertex detector



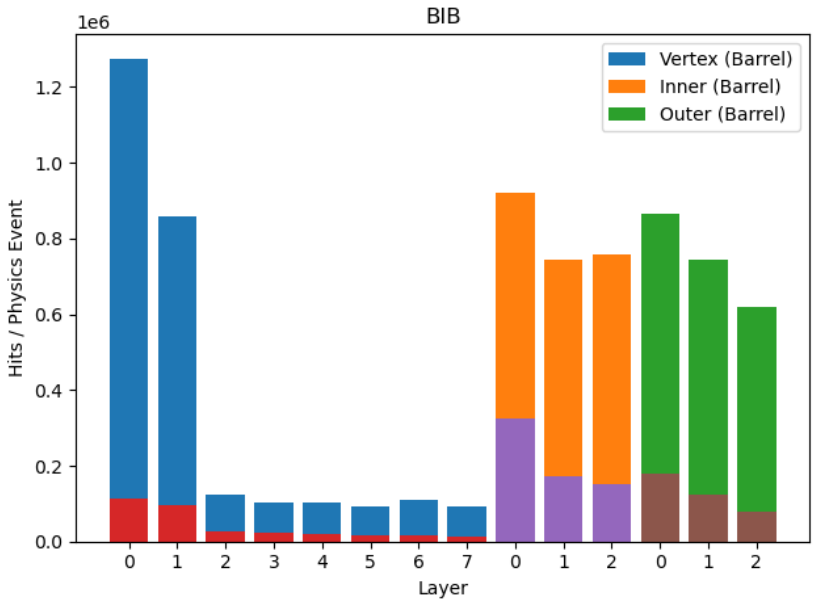
Seed 1: Skip high occupancy inner layer

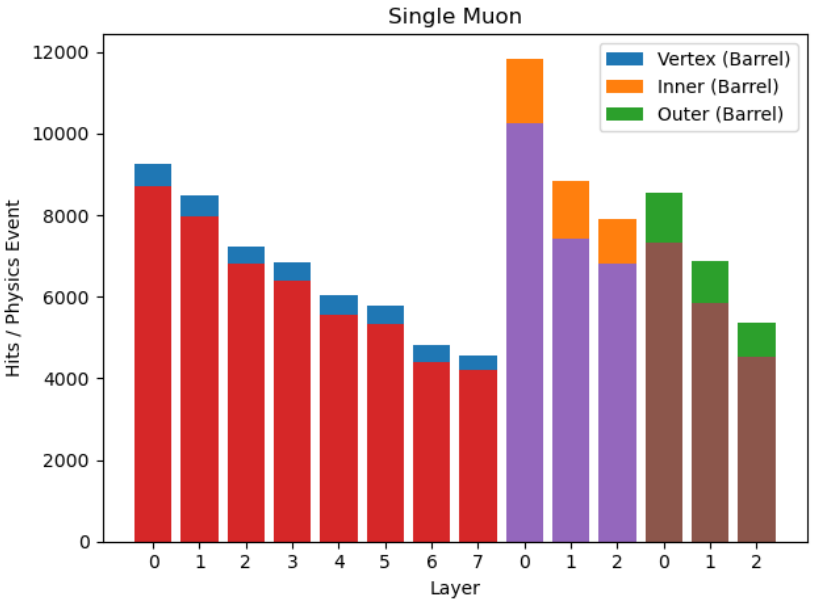


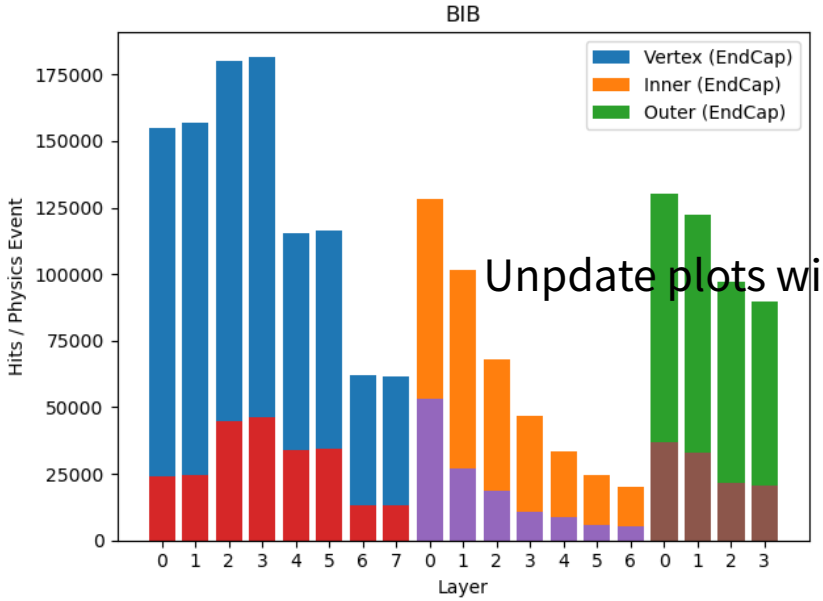
- **Using only inner part of the Vertex doubles**
  - Prevents redundant “too close together” combinations
  - Future: Reduce hits with doublet requirements in double layer?
- **Seed 1 reduces combinations by avoiding innermost layer**
  - Keeps inner endcap for coverage, occupancy high only at small R

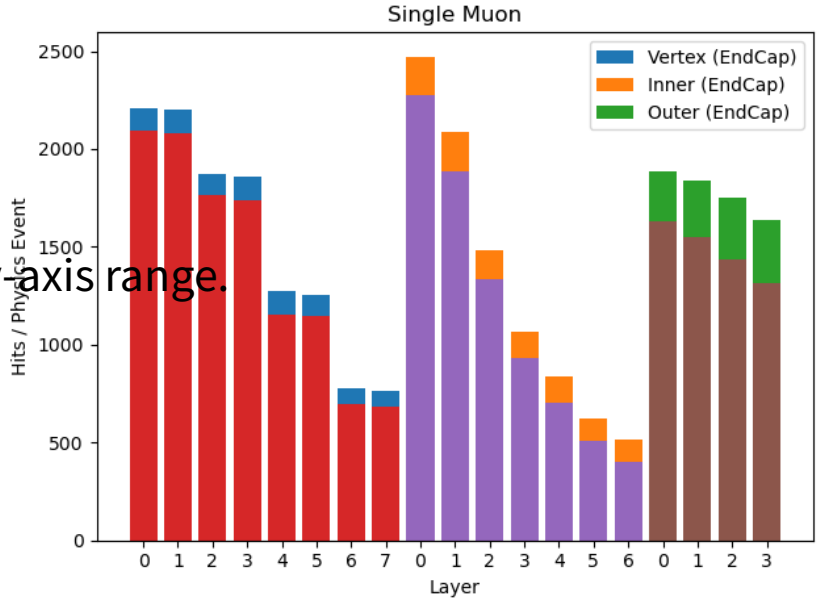
BIB Distribution

Second color is number of hits after timing cuts.



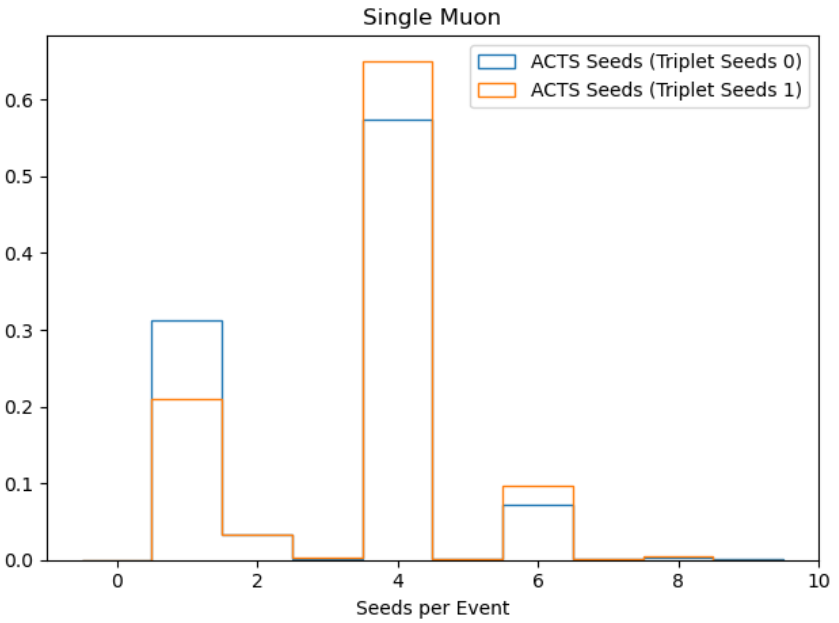






Unpdate plots with same y-axis range.

Found Seeds in Full BIB



- Assume hit in all 4 layers
  - 3 choose 4 = 4
- Missing seeds at low p_T
- Same efficiency in both